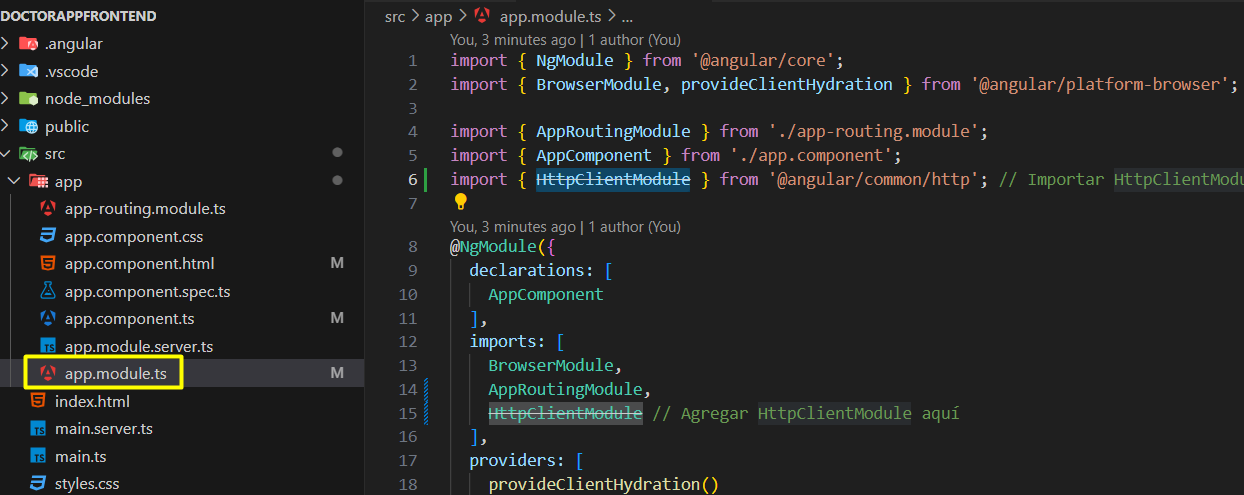
# HttpClientModule (app.module.ts)

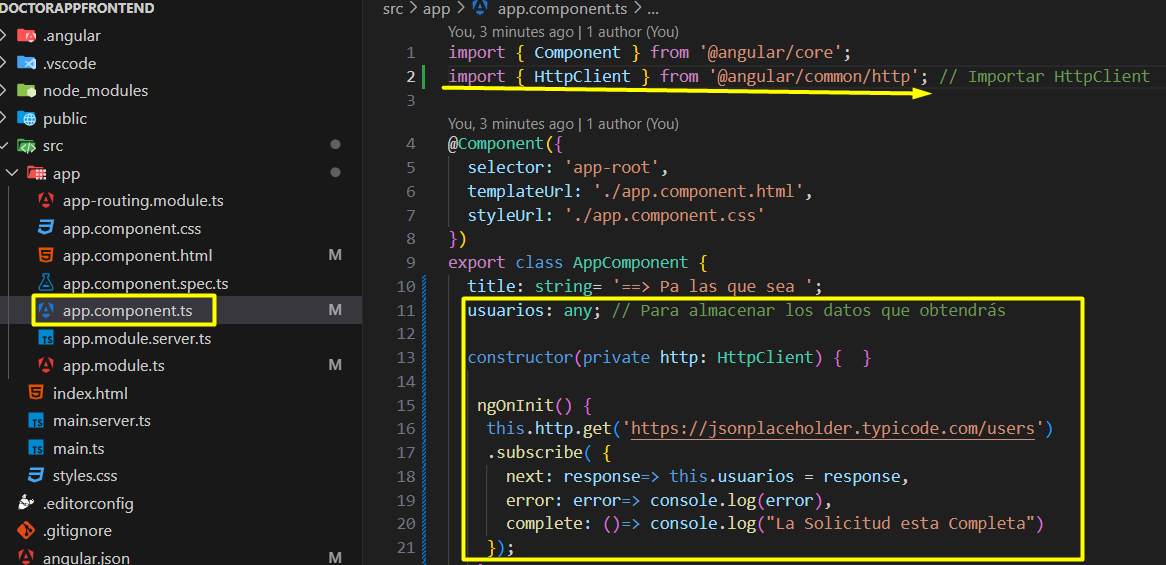
import { HttpClientModule } from '@angular/common/http';

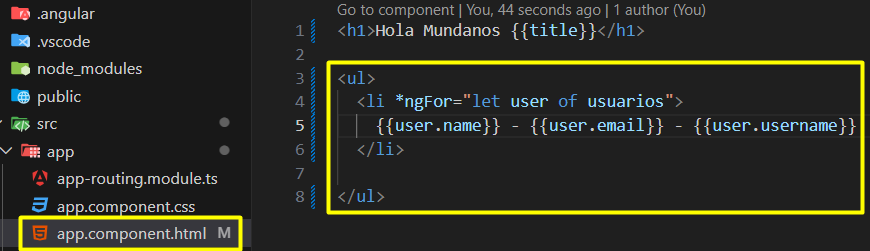


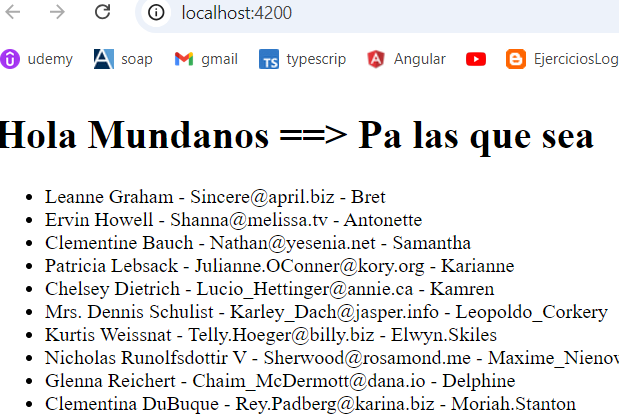
## Componentes (app.component.ts)

<https://jsonplaceholder.typicode.com/users> lista de usuarios

<https://jsonplaceholder.typicode.com/posts> otros





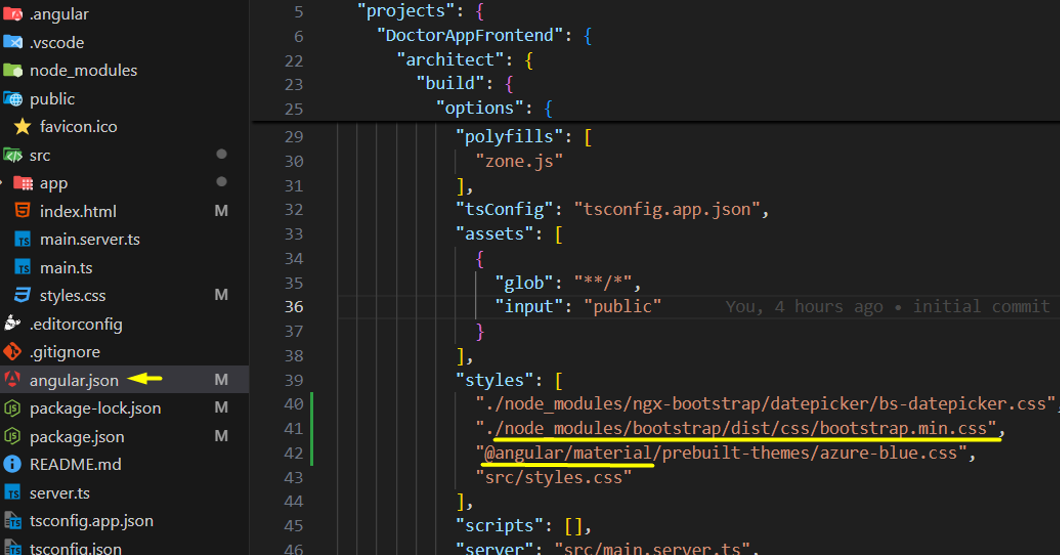


## Agregar Material y BoopTrap

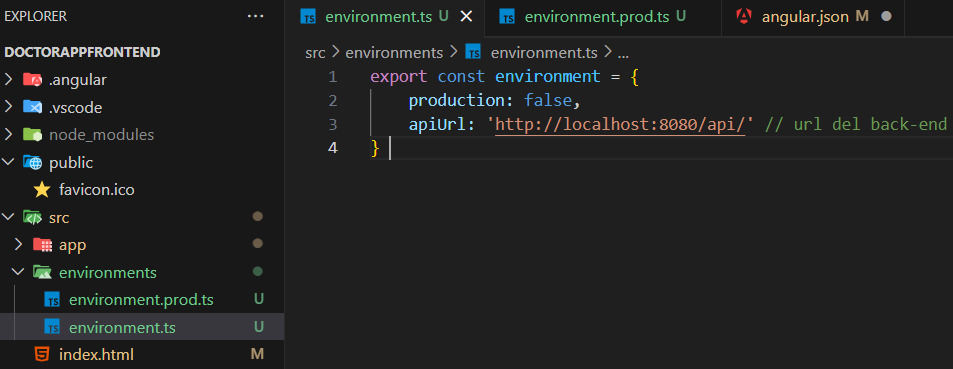
ng add @angular/material

ng add ngx-bootstrap

npm install bootstrap @popperjs/core



# CREAR ARCHIVOS ENVIRONMENTS



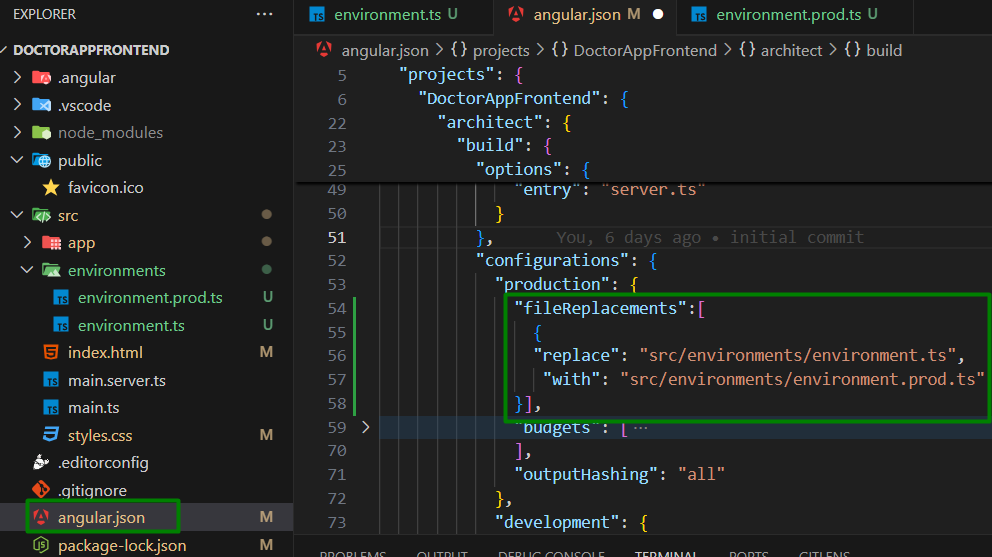
export const environment = {

    production: false, // para produccion es true

    apiUrl: 'http://localhost:8080/api/' // url del back-end

}

**Angular.json**



 "fileReplacements":[ {

                "replace": "src/environments/environment.ts",

                 "with": "src/environments/environment.prod.ts"

              }],

# CREACION DE MODULO

🡺 ng generate module compartido

🡺 ng g m material

🡺 ng g m usuario

## Compartir Módulos

import { ReactiveFormsModule, FormsModule } from '@angular/forms';

import { HttpClientModule } from '@angular/common/http';

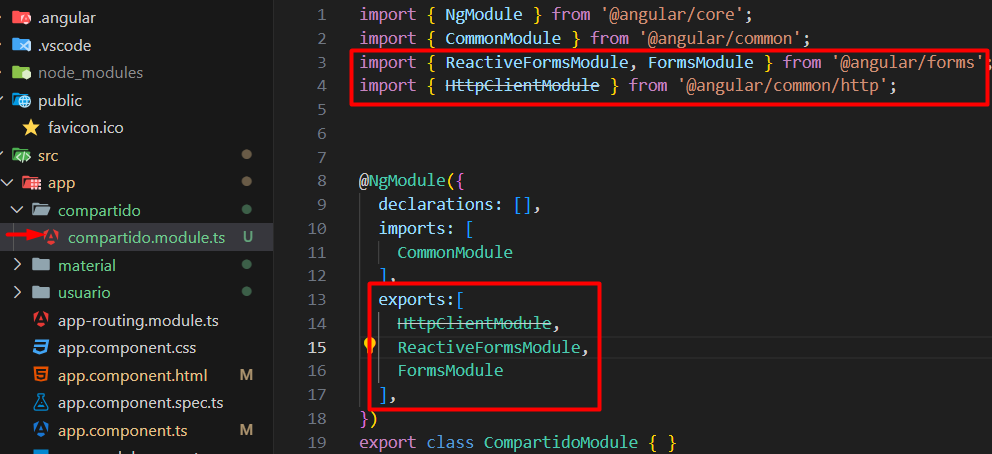
exports:[

    HttpClientModule,

    ReactiveFormsModule,

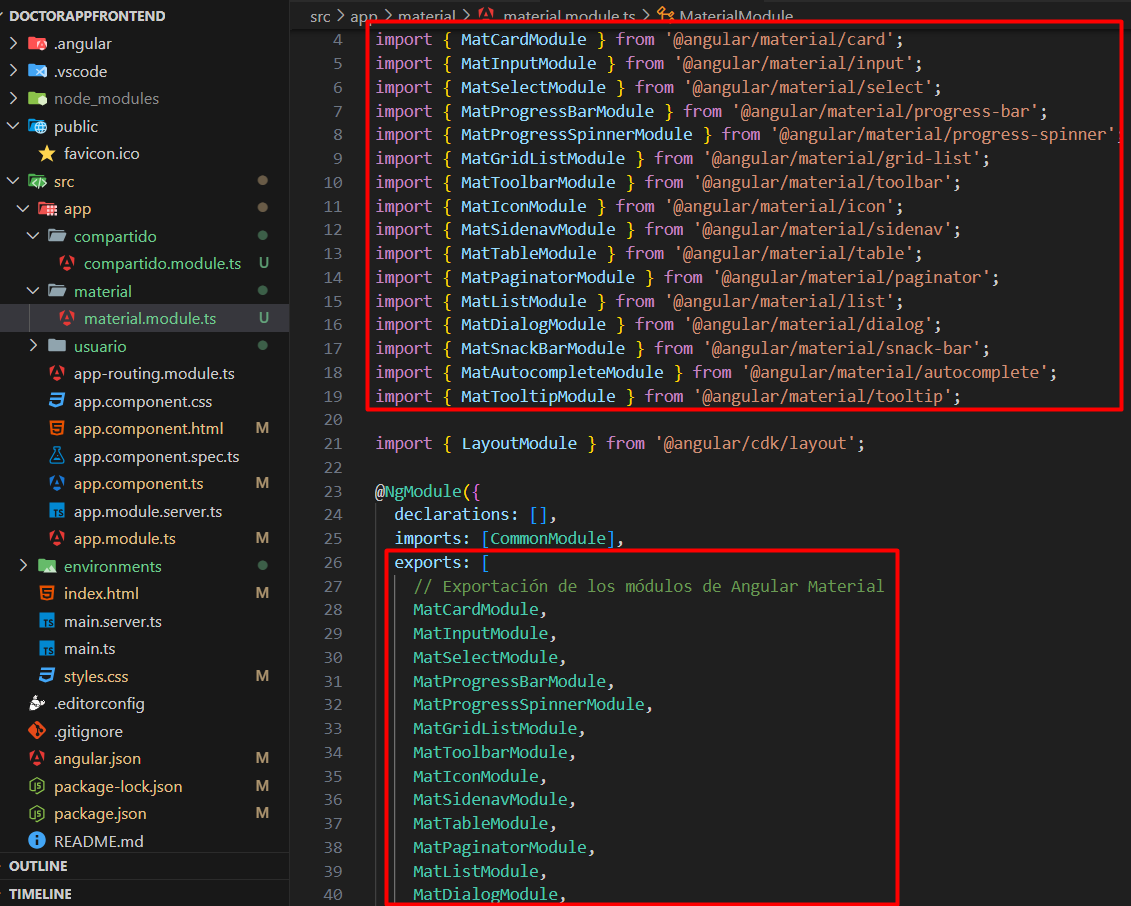
    FormsModule

  ],



## **MATERIAL MODULES**

import {MatSidenavModule} from '@angular/material/sidenav';



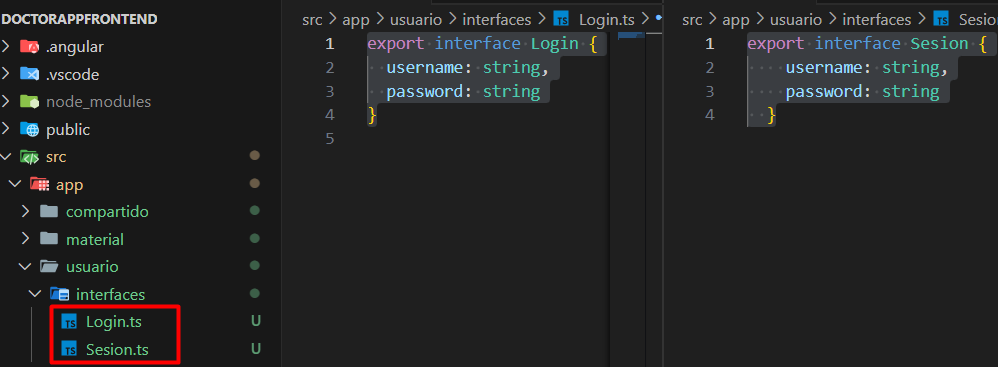
# CREAR INTERFSACES Y SERVICIOS

export interface Sesion {

    username: string,

    password: string

  }



# Generarar componente

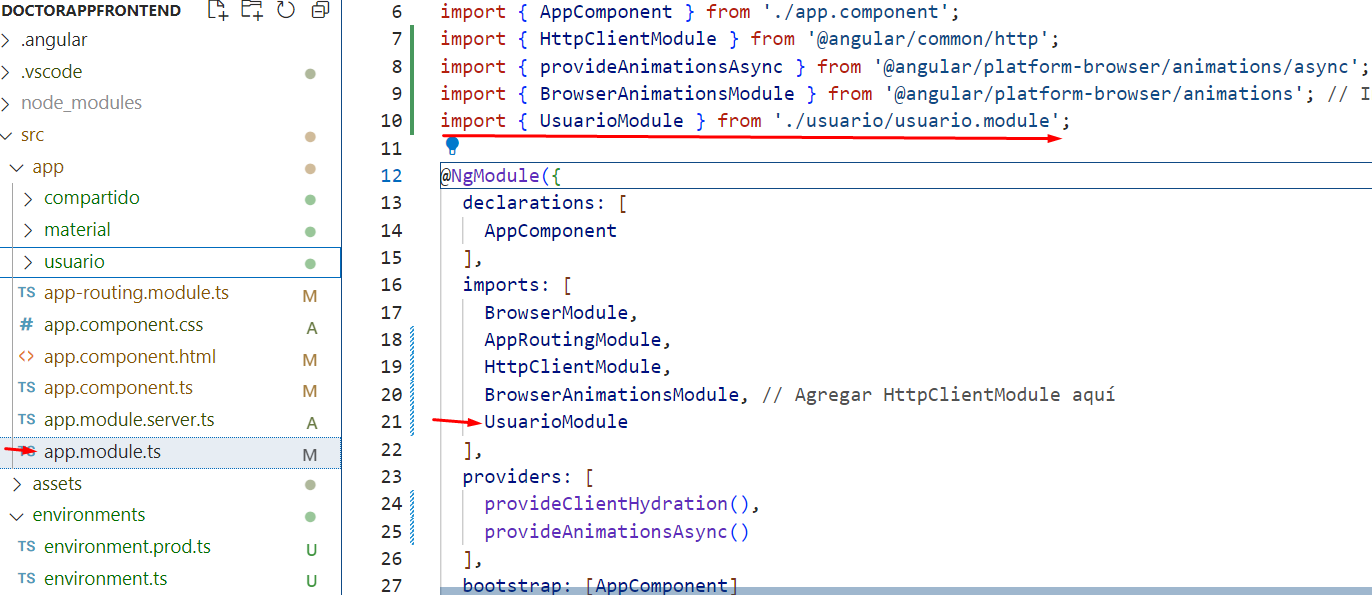
ng g s usuario/servicios/usuario

 ng g c compartido/layout  
 ng g c compartido/Dashboard   
 ng g c usuario/login   
 ng g s compartido/compartido

# Modulos



El usuario modulo se puede mandar al app.modulo principal

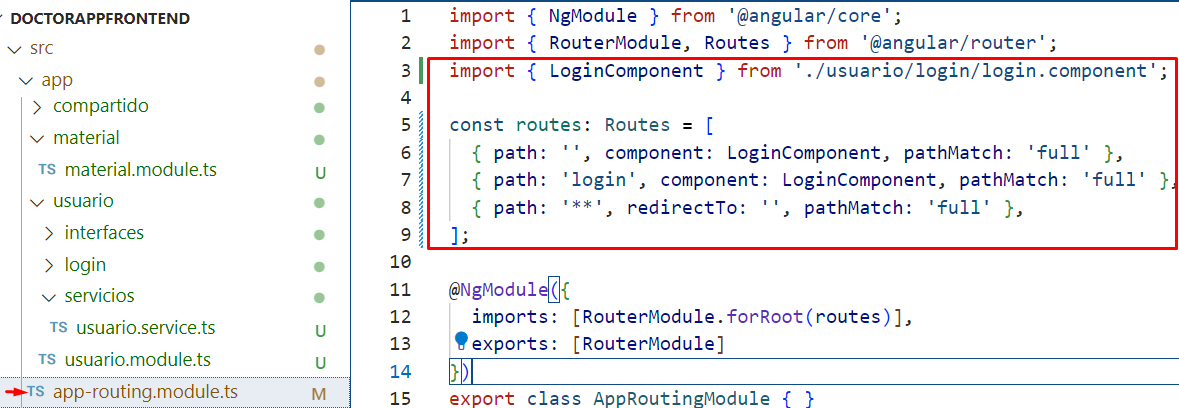


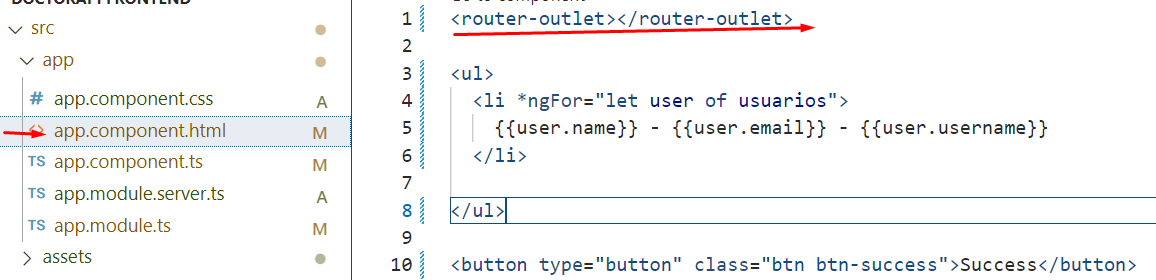
# Servicios

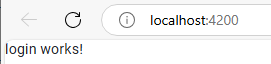


# ENRUTAMIENTO

Empieza por el Login

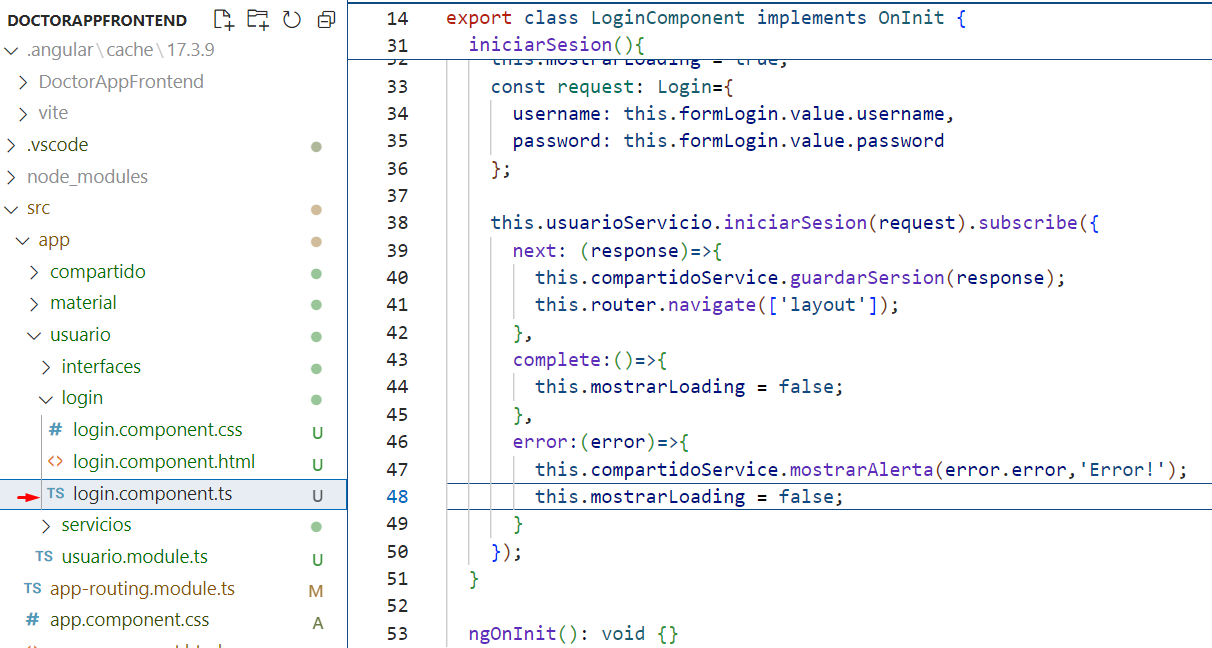




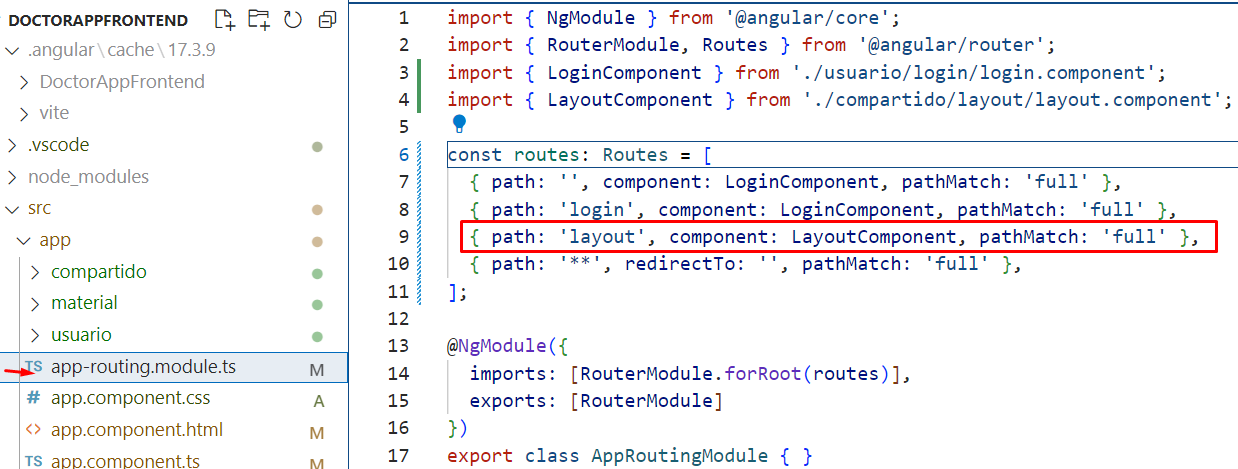


# UTILIZAR USERVICE PARA LOGIN

# 



## Agregar Ruta



# MEJORAR LOGIN

